

Kierunek: Informatyka Algorytmiczna (INA)

Specjalność: -

PRACA DYPLOMOWA
INŻYNIERSKA

**Implementacja programu GRAPHPLAN do
planowania akcji z wykorzystaniem
programowania ograniczeń**

Radosław Wojtczak

Opiekun pracy
dr Przemysław Kobylański

Słowa kluczowe:

Planowanie

Grafy

Programowanie ograniczeń

Sztuczna inteligencja

Streszczenie

Obiektem badań poniższej pracy jest metodologia planowania o nazwie **"Graphplan"**, której esencją jest wykorzystanie własności grafów skierowanych w trakcie ustalania optymalnego planu transformacji stanu początkowego w stan końcowy w ustalonej przestrzeni przy wykorzystaniu wcześniej zdefiniowanych operatorów.

Dodatkowym aspektem pracy jest użycie programowania ograniczeń w celu zwiększenia wydajności jak i zmniejszenia przestrzeni poszukiwań przez algorytm w trakcie generowania planu.

Ów praca składa się z formalnego opisu przytoczonego algorytmu, przedstawienia przykładów zastosowania, implementacji, której wynikiem jest graf, przedstawiający optymalny plan wykonywanych operacji oraz zmian, jakie dzięki nim wykonujemy w świecie, omówienie opcjonalnych rozszerzeń, które w zależności od sytuacji mogą wpłynąć na efektywność algorytmu oraz przeprowadzonych testów, których zadaniem jest wskazanie mocnych, jak i słabych stron przedmiotu badań.

Grafy są generowane przy pomocy programu, który opiera się o moduł **graphviz**, dostępny w języku programowania **python**.

Abstract

Tutaj treść streszczenia po angielsku.

Spis treści

Spis rysunków	III
Spis tabel	IV
Wstęp	1
1 Wprowadzenie	2
1.1 Planowanie	2
1.2 Planowanie przy użyciu komputerów	2
1.3 Użycie grafów- motywacja	2
2 GRAPHPLAN	3
2.1 Wprowadzenie	3
2.2 Warunki początkowe	3
2.3 Warstwy grafu	3
2.4 Typy operatorów	3
2.5 Wzajemne wykluczanie	3
2.6 Wyszukiwanie planu	3
2.7 Prosty przykład	3
3 Programowanie ograniczeń	4
3.1 Wprowadzenie	4
3.2 Wykorzystanie w programowaniu w logice	4
3.3 Obrazowe przykłady	4
4 Implementacja	5
4.1 Połączenia między komponentami	5
4.2 Implementacja algorytmu	5
4.3 Generowanie grafów	5
4.4 Interfejs użytkownika	5
5 Instalacja i wdrożenie	6
5.1 Instalacja pakietu SWI-Prolog	6
5.2 Instalacja języka Python	6
5.3 Sposób uruchomienia programu	6
6 Testy	7
6.1 15	8
6.1.1 Wprowadzenie	8
6.1.2 Przykład	8
6.1.3 Szczegóły implementacyjne	8
6.1.4 Wyniki	8
6.1.5 Wnioski	8
6.2 CargoBot	8
6.2.1 Wprowadzenie	8

6.2.2	Przykład	8
6.2.3	Szczegóły implementacyjne	8
6.2.4	Wyniki	8
6.2.5	Wnioski	8
6.3	Przemieszczanie w przestrzeni	8
6.3.1	Wprowadzenie	8
6.3.2	Przykład	8
6.3.3	Szczegóły implementacyjne	8
6.3.4	Wyniki	8
6.3.5	Wnioski	8
6.4	Wieża Hanoi	8
6.4.1	Wprowadzenie	8
6.4.2	Przykład	8
6.4.3	Szczegóły implementacyjne	8
6.4.4	Wyniki	8
6.4.5	Wnioski	8
6.5	Lis, Gęś i Ziarno	8
6.5.1	Wprowadzenie	8
6.5.2	Przykład	8
6.5.3	Szczegóły implementacyjne	8
6.5.4	Wyniki	8
6.5.5	Wnioski	8
6.6	Problem komiwojażera	8
6.6.1	Wprowadzenie	8
6.6.2	Przykład	8
6.6.3	Szczegóły implementacyjne	8
6.6.4	Wyniki	8
6.6.5	Wnioski	8
6.7	Osiem Hetmanów	8
6.7.1	Wprowadzenie	8
6.7.2	Przykład	8
6.7.3	Szczegóły implementacyjne	8
6.7.4	Wyniki	8
6.7.5	Wnioski	8
Podsumowanie		9
Bibliografia		10
A Zawartość płyty CD		11

Spis rysunków

Spis tabel

Wstęp

Celem pracy jest zaimplementowanie algorytmu do planowania akcji o nazwie **GRAPHPLAN**, który po raz pierwszy został sformalizowany i opisany w pracy pod tytułem **"Fast Planning Through Planning Graph Analysis"**[1] przez Panów: Avrima L. Blum'a i Merricka L. Furst'a. Praca składa się z szczęściu rozdziałów.

W rozdziale pierwszym poruszono aspekty historyczne odnośnie planowania akcji przy użyciu komputerów oraz jaką rolę pełni w niej GRAPHPLAN, dokonano teoretycznego porównania algorytmu względem nowoczesnych metod z przytoczonej dziedziny informatyki. Ponadto przedstawiono dlaczego w naturalny sposób do planowania akcji przedsięwzięto grafy.

W rozdziale drugim poddano dogłębną analizę implementowany algorytm- dokładnie opisano jego strukturę, warstwy, z których się składa oraz własności, które wyróżniają go na tle innych rozwiązań problemów związanych z planowaniem. W celu łatwiejszego przyswojenia mechanizmów stojących za GRAPHPLAN'em w trakcie opisu wprowadzono liczne proste przykłady wraz z grafikami wygenerowanymi przy pomocy narzędzi stworzonych na potrzeby ów pracy.

W rozdziale trzecim rozwinęto pojęcie programowania ograniczeń, wprowadzając formalną definicję, podstawowe słownictwo niezbędne do zrozumienia idei stojącej za tym sposobem programowania, omówiono benefity płynące z wykorzystania tego podejścia oraz przedstawiono obrazowo schemat funkcjonowania na podstawie prostych przykładów.

Rozdział czwarty skupia się na szczegółach implementacyjnych: wybranych językach programowania oraz technologiach wykorzystywanych również w warstwach graficznych programu. Dokonano szczegółowego opisu interfejsu użytkownika oraz jego możliwości, połączeń między komponentami oraz ważniejszych funkcji stanowiących trzon pracy.

Rozdział piąty przedstawia sposób instalacji oraz instrukcję obsługi programu, dodatkowo zawiera instrukcję odnośnie instalowania wszystkich niezbędnych komponentów wykorzystywanych w pracy, w których skład wchodzi interpretery jak i kompilatory używanych języków programowania oraz wszystkie biblioteki i moduły.

Rozdział szósty przedstawia przeprowadzone testy, które badają możliwości algorytmu w wcześniej spreparowanych środowiskach. W tej części została przeprowadzona analiza wydajnościowa algorytmu, weryfikacja wygenerowanych planów pod względem poprawności oraz porównanie otrzymanych wyników z innymi powszechnie wykorzystywanymi metodami planowania. Każdy z testów zawiera w sobie wniosek, w którym odbywa się zbiorcza ocena wszystkich wyżej wymienionych aspektów.

Końcowy rozdział stanowi zbiorcze podsumowanie pracy z komentarzem odnośnie potencjalnych rejonów, w których algorytm mógłby znaleźć swoje zastosowanie.

Rozdział 1

Wprowadzenie

1.1 Planowanie

Codziennie ludzie używają słowa "plan" w różnych kontekstach: firma planuje rozkład palet na magazynie, trener planuje strategię na najbliższy mecz, student planuje jak rozwiązać zadanie na kolokwium, czy chociażby człowiek codziennie planuje listę zakupów.

Mimo różnych kontekstów istnieją wyodrębnione wspólne cechy każdego z planów:

Definicja 1.1 *Warunki początkowe* - stan świata przed zastosowaniem jakichkolwiek akcji. W dalszej części pracy również określane jako *stany początkowe*.

- Każdy plan musi mieć jasno zadeklarowane warunki początkowe. Dzięki dokładnej wiedzy o świecie możliwym jest poprawne określenie akcji, przy pomocy których wprowadzane są modyfikacje obecnego stanu aż do otrzymania zadowalających rezultatów. Dla przykładu, firma musi wiedzieć ile oraz jakie palety przybędą na magazyn zanim rozpocznie planowanie rozkładu dostawy na magazynie.

Definicja 1.2 *Akcja* - działanie zmieniające przedstawiony świat w ściśle określony sposób. W dalszej części pracy również określane jako *operator*.

- Akcje - akcje pozwalają na modyfikację przedstawionego świata. Każda z akcji składa się z podmiotu, na który działa oraz czynności, która jest względem wskazanego podmiotu wykonywana. Przykładem dobrze określonej akcji może być przeniesienie klocka z jednego stolika na drugi- składa się ona z podmiotu w postaci klocka, oraz czynności w postaci przenoszenia, które możemy traktować jako ruch. Czynności mogą różnić się od siebie w kwestii skomplikowania, najważniejszym jest, aby były określone poprawnie, co sprowadza się do tego, aby były wykonalne w zdefiniowanym świecie..

Definicja 1.3 *Cel* - Oczekiwany stan świata.

- Cele - kwintesencją każdego planu jest cel, który należy uzyskać. Zwyczajowo plany składają się z celów możliwych do osiągnięcia ze stanu początkowego przy pomocy zdefiniowanej operacji, jednakże trzeba wziąć pod uwagę sytuację, w której niemożliwym jest uzyskanie wskazanego celu, szczególnie próbując automatyzować pojęcie planowania.

Przy pomocy powyższych definicji możliwym jest sformalizowanie pojęcia stojącego za słowem **plan**.

Definicja 1.4 *Plan*- lista akcji, której zastosowanie do stanu początkowego powoduje jego zmianę do stanu określonego w ramach celu.

1.2 Planowanie przy użyciu komputerów

1.3 Użycie grafów- motywacja

Rozdział 2

GRAPHPLAN

- 2.1 Wprowadzenie
- 2.2 Warunki początkowe
- 2.3 Warstwy grafu
- 2.4 Typy operatorów
- 2.5 Wzajemne wykluczanie
- 2.6 Wyszukiwanie planu
- 2.7 Prosty przykład

Rozdział 3

Programowanie ograniczeń

3.1 Wprowadzenie

3.2 Wykorzystanie w programowaniu w logice

3.3 Obrazowe przykłady

Rozdział 4

Implementacja

4.1 Połączenia między komponentami

4.2 Implementacja algorytmu

4.3 Generowanie grafów

4.4 Interfejs użytkownika

Rozdział 5

Instalacja i wdrożenie

5.1 Instalacja pakietu SWI-Prolog

5.2 Instalacja języka Python

5.3 Sposób uruchomienia programu

Rozdział 6

Testy

6.1 15

6.1.1 Wprowadzenie

6.1.2 Przykład

6.1.3 Szczegóły implementacyjne

6.1.4 Wyniki

6.1.5 Wnioski

6.2 CargoBot

6.2.1 Wprowadzenie

6.2.2 Przykład

6.2.3 Szczegóły implementacyjne

6.2.4 Wyniki

6.2.5 Wnioski

6.3 Przemieszczanie w przestrzeni

6.3.1 Wprowadzenie

6.3.2 Przykład

6.3.3 Szczegóły implementacyjne

6.3.4 Wyniki

6.3.5 Wnioski

6.4 Wieża Hanoi

6.4.1 Wprowadzenie

6.4.2 Przykład

6.4.3 Szczegóły implementacyjne

6.4.4 Wyniki

6.4.5 Wnioski

Podsumowanie

W podsumowaniu należy określić stan zakończonych prac projektowych i implementacyjnych. Zaznaczyć, które z zakładanych funkcjonalności systemu udało się zrealizować. Omówić aspekty pielęgnacji systemu w środowisku wdrożeniowym. Wskazać dalsze możliwe kierunki rozwoju systemu, np. dodawanie nowych komponentów realizujących nowe funkcje.

W podsumowaniu należy podkreślić nowatorskie rozwiązania zastosowane w projekcie i implementacji (niebanalne algorytmy, nowe technologie, itp.).

Bibliografia

- [1] Fast planning through planning graph analysis. Web pages: <https://www.cs.cmu.edu/~avrim/Papers/graphplan.pdf> [ostatni dostęp: 21.10.2022].

Załącznik A

Zawartość płyty CD

W tym rozdziale należy krótko omówić zawartość dołączonej płyty CD.